

The table and related tags

The table tag displays information in rows and columns.

The following code:

```
<table>
<tr>
  <td>Row 1 - Col 1</td>
  <td>Row 1 - Col 2</td>
</tr>
<tr>
  <td>Row 2 - Col 1</td>
  <td>Row 2 - Col 2</td>
</tr>
</table>
```

Produces a table that looks like:

Row 1 - Col 1 Row 1 - Col 2

Row 2 - Col 1 Row 2 - Col 2

Inside the **<table>** tag are **<tr>** or table row tags which in turn contain **<td>** or table cell tags.

Data goes in the **<td>** cell. That data can be text, other tags or whatever. Remember that first comes **<table>**, then **<tr>**, then **<td>**, then *and only then*, data.

Below I added some **CSS** and the border attribute (border="1") to better display the rows and columns.

Row 1 - Col 1	Row 1 - Col 2
Row 2 - Col 1	Row 2 - Col 2

Here's the rub. There has to be the same number of cells in each row *unless* you indicate a table cell spans multiple columns and/or rows.

Multiple columns:

```
<table border="1">
<tr>
  <td colspan="2">Row 1 - Col 1 & Col 2</td>
</tr>
<tr>
  <td>Row 2 - Col 1</td>
  <td>Row 2 - Col 2</td>
</tr>
</table>
```

Row 1 - Col 1 & Col 2	
Row 2 - Col 1	Row 2 - Col 2

Multiple rows:

```
<table border="1">
  <tr>
    <td>Row 1 - Col 1</td>
    <td rowspan="2">Row 1 & Row 2 - Col 2</td>
  </tr>
  <tr>
    <td>Row 2 - Col 1</td>
  </tr>
</table>
```

Row 1 - Col 1	Row 1 & Row 2 - Col 2
Row 2 - Col 1	

The strong and b elements

Here I introduce semantic HTML

Take the following:

This bold text is in strong tags.

This bold text is in b tags.

``This bold text`` is in strong tags.

``This bold text`` is in b tags.

They both make text bold. In terms of presentation they are identical. So why have two elements that do the same thing?

They may look the same to humans, but the web crawlers or bots – that crucial second audience – see something quite different. When a search engine spiders and analyzes a page, text in `` tags is considered important. Text in `` tags is not.

Say you are making a page on "Digitgradient". I can't imagine why you would want to, but if you did you might include the following:

Digitgradient

Verb

– To walk on your toes like a dog or cat.

By putting "Digitigradient" in tags you are telling the crawlers that "Digitigradient" is important to the meaning of the page while "verb", in tags, is not. Placing "Digitigradient" in strong tags gives it "strong" semantic weight.

The dictionary defines semantics as "the study of meaning". On the web semantics puts meaning into HTML – meaning search engines can use to evaluate a page's SEO.

Semantics is not new to HTML5. Web designers have always had a degree of control of what the bots did and didn't judge significant. For example, anything in <h1> tags has been considered more important than something in say <h6> tags, but HTML5 takes semantics to a whole new level.

My guess is that with time, usage will establish what means what. Incorporating semantics into HTML5 is a good idea – an idea that needs to mature and develop, but a very good idea nonetheless.

The a or anchor tag

Hyperlinks, or **links**, are how you move around the web. An a or anchor tag is how you make hyperlinks in HTML.

When you click on a link your computer (the client) asks another computer (the **server**) to send it a file. When all is said and done that's all that is really going on. How all that works is complicated, but *what happens* – what is actually going on – is not.

Always keep in mind the fact that *everything* on the net is stored as files on computers connected to the internet. There are "local" files, or files on your computer, and then there are billions of files on **servers** all over the planet. Your computer knows that the file "a-tag.html" is a webpage because it has an ".html" extension and opens it in your browser for your viewing pleasure.

The code for the link (**a-tag**) to this page looks like this:

```
<a href="http://www.google.com">a tag</a>
```

If no file name is included in the **URL** it is assumed to be the **index** or home page.

```
<a href="http:// www.google.com">XYZ</a>
```

Any text or image (we'll get into that later), can go where I wrote "XYZ". The "XYZ" – what is inside the a tag – has no effect on where the link will take you. It does however tell visitors where they will go if they click the link.